

CENTRO FEDERAL DE EDUCAÇÃO TECNOLÓGICA DE MINAS GERAIS
ENGENHARIA DE COMPUTAÇÃO
PROGRAMAÇÃO ORIENTADA A OBJETOS
PROF. DR. EDUARDO HABIB BECHELANE MAIA

GUILHERME ALVARENGA DE AZEVEDO

DOCUMENTAÇÃO DE SISTEMA:
Consultório Médico

Divinópolis – MG
2023

SUMÁRIO

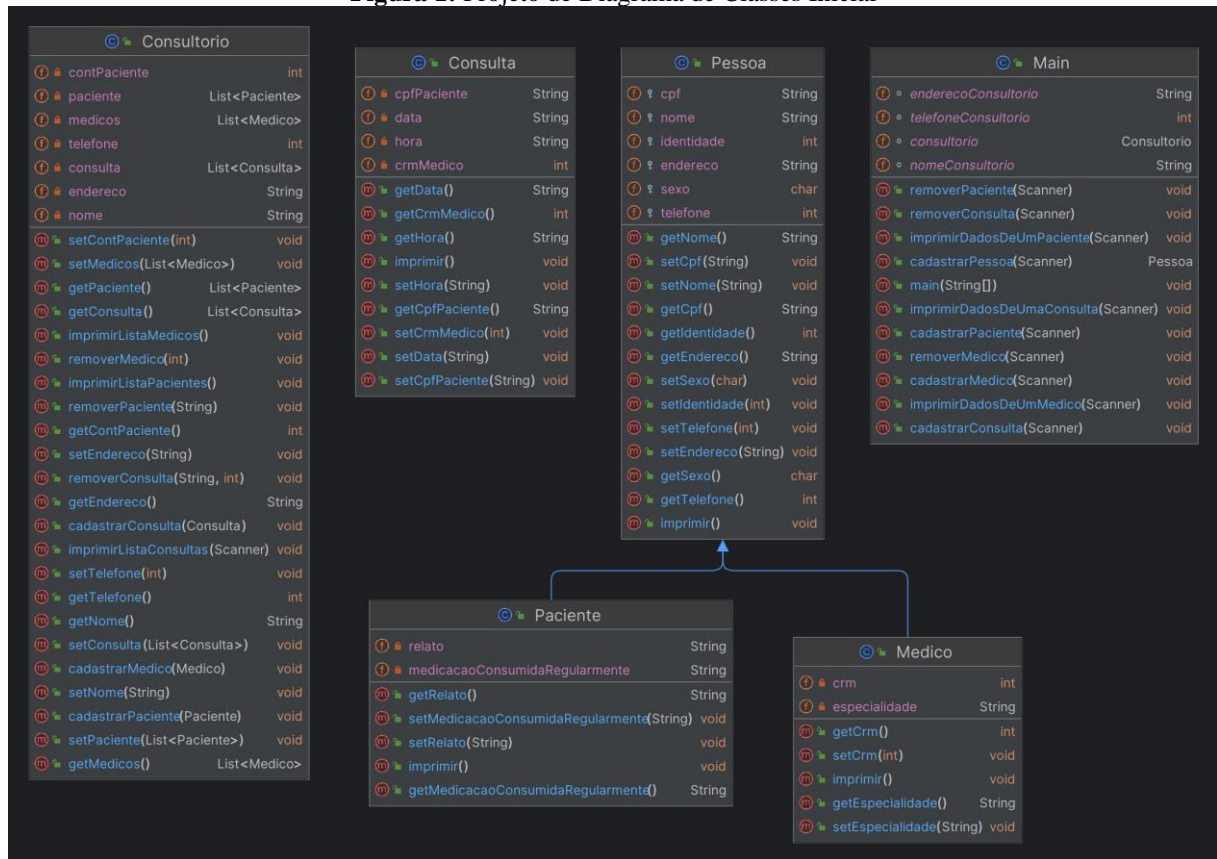
1. INTRODUÇÃO.....	2
2. DECISÕES DE IMPLEMENTAÇÃO.....	3
3. CASOS DE TESTE.....	5
4. RESULTADOS DOS TESTES REALIZADOS.....	6
5. ANÁLISE DOS RESULTADOS.....	14

1. INTRODUÇÃO

Inicialmente foi realizada uma profunda análise da proposta de desenvolvimento de um sistema para um consultório médico. A principal ideia era de que o sistema fosse capaz de armazenar e organizar os dados relacionados aos principais elementos/autores de um determinado consultório médico, isto é, as informações dos pacientes, dos médicos e das consultas marcadas e realizadas no consultório.

Para cada um destes elementos foram propostas as principais características e ações que deveriam ser possíveis de serem realizadas no sistema, além de elementos adicionais para facilitar na compreensão da aplicação. Suas implementações gerais seguem o modelo do seguinte projeto de diagrama de classes:

Figura 1: Projeto de Diagrama de Classes Inicial



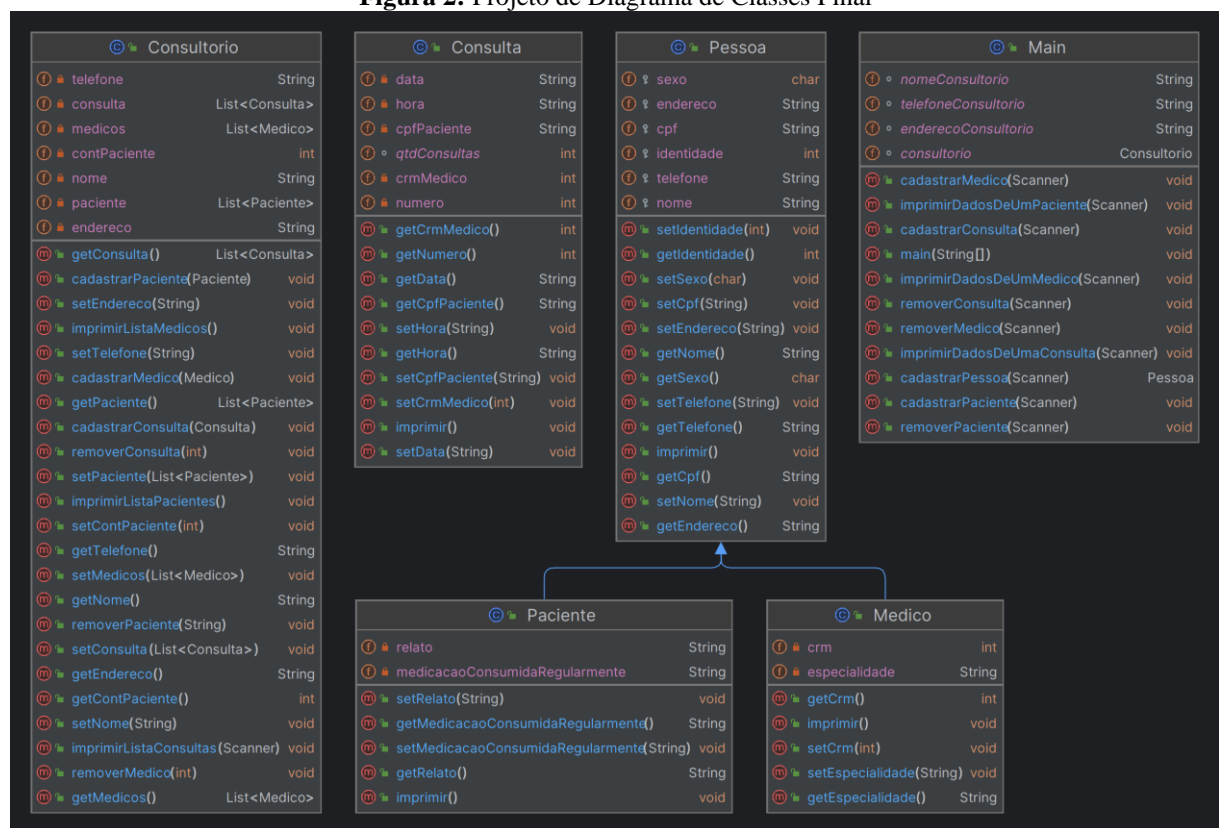
Fonte: IntelliJ IDEA - Autor (2023)

2. DECISÕES DE IMPLEMENTAÇÃO

Com a ideia geral já determinada, começaram as decisões de implementação pensando especificamente na programação do sistema. Inicialmente, o programa já teria pré-definido o nome, número de telefone e endereço do consultório, já que o sistema será utilizado sempre por um mesmo consultório.

Posteriormente, o projeto do diagrama de classes foi alterado e complementado, para que algumas implementações ficassem mais simplificadas.

Figura 2: Projeto de Diagrama de Classes Final



Fonte: IntelliJ IDEA – Autor (2023)

A primeira alteração realizada foi a mudança do tipo de variável dos atributos “telefone” nas classes “Pessoa” e “Consultorio”. Inicialmente foi proposto que fosse utilizado o tipo *int* (inteiro), porém, pelo fato de esse tipo de variável armazenar apenas valores até 2.147.483.647 (32 bits) e um número de telefone poder assumir um valor maior, foi optado por se utilizar o tipo *String* para resolver este problema.

Além disso, uma importante mudança foi a adição do atributo privado “numero” e do atributo estático “qtdConsultas”, para ter um controle melhor da quantidade de consultas já marcadas no consultório e para facilitar a identificação de cada consulta através de um número único. Com isso, o método de remover consulta foi alterado para que receba apenas o número

da consulta. A principal razão para realizar essa mudança, foi pelo fato de que um mesmo conjunto médico-paciente pode ter mais de uma consulta marcada no sistema. Então, a identificação única implementada facilita nessa distinção.

Após essas predefinições, foram decididas que a linguagem de programação, Java, e a IDE (do inglês, Ambiente de Desenvolvimento Integrado), IntelliJ IDEA seriam utilizadas. As escolhas foram baseadas puramente no gosto do desenvolvedor e na facilidade (parcial) que a linguagem e a ferramenta trazem, tais como a geração de métodos automaticamente e criação e edição de diagramas de classes.

Dessa forma, como Java é prioritariamente Orientada a Objetos, os conceitos deste paradigma (abstração, encapsulamento, herança e polimorfismo) foram bastante utilizados no código para que houvesse sentido no desenvolvimento da aplicação, como pode ser visto abaixo marcado de vermelho:

Figura 3: Exemplificação de parte da Orientação a Objetos

```
public class Paciente extends Pessoa {
    4 usages
    private String relato;
    4 usages
    private String medicacaoConsumidaRegularmente;

    8 usages
    @Override
    public void imprimir() {
        super.imprimir();
        System.out.println("Relato: " + this.relato);
        System.out.println("Medicação: " + this.medicacaoConsumidaRegularmente);
    }

    8 usages
    public Paciente(String nome, char sexo, String endereco, String cpf, String telefone, int identidade, String relato,
        String medicacaoConsumidaRegularmente) {
        super(nome, sexo, endereco, cpf, telefone, identidade);
        this.relato = relato;
        this.medicacaoConsumidaRegularmente = medicacaoConsumidaRegularmente;
    }
}
```

Fonte: IntelliJ IDEA – Autor (2023)

3. CASOS DE TESTE

Nesta seção são listadas todas as funcionalidades do sistema que foram testadas.

- **Caso 01 - Menu de Opções**
- **Caso 02 - Cadastrar Médicos**
- **Caso 03 - Imprimir Dados de um Médico**
- **Caso 04 - Cadastrar Pacientes**
- **Caso 05 - Imprimir Dados de um Paciente**
- **Caso 06 – Cadastrar Pessoas com o mesmo CPF, identidade ou CRM**
- **Caso 07 - Cadastrar Consultas**
- **Caso 08 - Imprimir Consultas Cadastradas**
- **Caso 09 - Imprimir Consulta por Número**
- **Caso 10 - Remover Médico por CRM**
- **Caso 11 - Remover Paciente por CPF**
- **Caso 12 - Imprimir Consulta após Remoções**
- **Caso 13 - Remover Consulta por Número**
- **Caso 14 - Imprimir Informações sem Cadastro: Médicos**
- **Caso 15 - Imprimir Informações sem Cadastro: Pacientes**
- **Caso 16 - Imprimir Informações sem Cadastro: Consulta**
- **Caso 17 - Cadastrar Consulta sem Médicos e/ou Pacientes Cadastrados**
- **Caso 18 - Sair do Programa**

4. RESULTADOS TESTES

Nesta seção são exploradas todas as funcionalidades do sistema e o que cada uma faz de acordo com os testes realizados.

4.1. Caso 01 - Menu de Opções

A primeira visão do sistema mostra um menu de opções para que o usuário possa escolher:

Figura 4: Menu Inicial

```
-----Escolha uma opção-----
1 - Cadastrar um Médico;
2 - Cadastrar um Paciente;
3 - Cadastrar uma Consulta;
4 - Imprimir Dados de um Médico;
5 - Imprimir Consultas;
6 - Imprimir Dados do Paciente;
7 - Imprimir Consulta Específica;
8 - Remover um Médico;
9 - Remover um Paciente;
10 - Remover uma Consulta;
11 - Sair;
-----
```

Fonte: Autor (2023)

4.2. Caso 02 - Cadastrar Médicos

A primeira opção do sistema permite que o usuário cadastre médicos no sistema:

Figura 5: Opção – Cadastrar Médicos

```
1
-----Cadastrando Médico-----
Nome:
Guilherme
Sexo (M ou F):
M
Endereço:
Rua Tal
CPF:
123
Telefone:
123
Identidade:
123
CRM:
123
Especialidade:
Pediatra
Médico cadastrado com sucesso!
APORTE ENTER PARA CONTINUAR
-----
```

Fonte: Autor (2023)

4.3. Caso 03 - Imprimir Dados de um Médico

Para mostrar que a funcionalidade funciona, a opção número 4 do sistema detalha o médico pelo seu CRM:

Figura 6: Opção – Listar Médico por CRM

```

4
-----Imprimindo Dados de Médico-----
Digite o CRM do Médico que você deseja ver as informações:
123
Médico encontrado:
Nome: Guilherme
Sexo: M
Endereco: Rua Tal
CPF: 123
Telefone: 123
Identidade: 123
CRM: 123
Especialidade: Pediatra
APERTE ENTER PARA CONTINUAR
-----

```

Fonte: Autor (2023)

4.4. Caso 04 - Cadastrar Pacientes

A segunda opção do sistema permite que o usuário cadastre pacientes no sistema:

Figura 7: Opção – Cadastrar Pacientes

```

2
-----Cadastrando Paciente-----
Nome:
Fulano de Tal
Sexo (M ou F):
M
Endereço:
Rua 3
CPF:
2
Telefone:
2
Identidade:
152
Relato do Paciente:
Dores na Barriga
Medicação Consumida Regularmente:
Dorflex
Paciente cadastrado com sucesso!
APERTE ENTER PARA CONTINUAR
-----

```

Fonte: Autor (2023)

4.5. Caso 05 - Imprimir Dados de um Paciente

Para mostrar que a funcionalidade funciona, a opção número 6 do sistema detalha o paciente pelo seu CPF:

Figura 8: Opção – Listar Paciente por CPF

```
6
-----Imprimindo Dados de Paciente-----
Digite o CPF do Paciente que você deseja ver as informações:
2
Paciente encontrado:
Nome: Fulano de Tal
Sexo: M
Endereco: Rua 3
CPF: 2
Telefone: 2
Identidade: 152
Relato:
Medicação: Dorflex
APERTE ENTER PARA CONTINUAR
-----
```

Fonte: Autor (2023)

4.6. Caso 06 – Cadastrar Pessoas com o mesmo CPF, identidade ou CRM

Ao tentar cadastrar pessoas com CPF, identidade ou CRM já cadastrados o sistema mostra as mensagens de erro como nas imagens abaixo:

Figura 9: Cadastrar CPF Existente

```
CPF:
2
Já existe uma pessoa cadastrada com este CPF!
CPF:
123
Já existe uma pessoa cadastrada com este CPF!
CPF:
4
```

Fonte: Autor (2023)

Figura 10: Cadastrar Identidade Existente

```
Identidade:
123
Já existe uma pessoa cadastrada com este número de Identidade!
Identidade:
152
Já existe uma pessoa cadastrada com este número de Identidade!
Identidade:
5
```

Fonte: Autor (2023)

Figura 11: Cadastrar CRM Existente

```

CRM:
123
Já existe um médico cadastrado com este CRM!
CRM:
6
Especialidade:
Psiquiatra

```

Fonte: Autor (2023)

Isso foi feito justamente para que não haja ambiguidade nos registros do sistema do consultório.

4.7. Caso 07 - Cadastrar Consultas

A terceira opção do sistema permite que o usuário cadastre consultas no sistema:

Figura 12: Opção – Cadastrar Consultas

```

3
-----Cadastrando Consulta-----
Data:
20/09/2024
Hora:
15
CPF do Paciente:
3
Não há pacientes registrados com esse CPF!
CPF do Paciente:
2
CRM do Médico:
3
Não há médicos registrados com esse CRM!
CRM do Médico:
123
Consulta cadastrada com sucesso!
APERTE ENTER PARA CONTINUAR
-----

```

Fonte: Autor (2023)

Na imagem acima, também é mostrada uma funcionalidade em que, caso o usuário digite um CPF ou CRM de registros inexistentes, não é possível cadastrar a consulta até indicar valores válidos.

4.8. Caso 08 - Imprimir Consultas Cadastradas

Para mostrar que a funcionalidade anterior funciona, a opção número 5 do sistema detalha as consultas cadastradas:

Figura 13: Opção – Listar Consultas

```
5
-----Imprimindo Lista de Consultas-----
Consulta 1
Número da Consulta: 1
Data: 20/09/2024
Hora: 15
CPF Paciente: 2
CRM Médico: 123
APERTE ENTER PARA CONTINUAR
-----
```

Fonte: Autor (2023)

4.9. Caso 09 - Imprimir Consulta por Número

A opção número 7 imprime uma consulta específica ao ser indicado seu número específico (mostrado na funcionalidade anterior - 5):

Figura 14: Opção – Imprimir Consulta por Número

```
7
-----Imprimindo Dados de Consulta-----
Digite o número da Consulta que você deseja ver as informações:
1
Consulta encontrada:
Número da Consulta: 1
Data: 20/09/2024
Hora: 15
CPF Paciente: 2
CRM Médico: 123
Paciente de CPF 2:
Nome: Fulano de Tal
Sexo: M
Endereco: Rua 3
CPF: 2
Telefone: 2
Identidade: 152
Relato:
Medicação: Dorflex
Médico de CRM 123:
Nome: Guilherme
Sexo: M
Endereco: Rua Tal
CPF: 123
Telefone: 123
Identidade: 123
CRM: 123
Especialidade: Pediatra
APERTE ENTER PARA CONTINUAR
```

Fonte: Autor (2023)

4.10. Caso 10 - Remover Médico por CRM

A opção número 8 solicita o CRM do médico do qual o registro será apagado:

Figura 15: Opção – Remover Médico por CRM

```

8
-----Removendo Médico-----
Digite o CRM do Médico que você deseja remover:
123
Médico removido com sucesso!
APORTE ENTER PARA CONTINUAR
-----

```

Fonte: Autor (2023)

4.11. Caso 11 - Remover Paciente por CPF

A opção número 9 solicita o CPF do paciente do qual o registro será apagado:

Figura 16: Opção – Remover Paciente por CPF

```

9
-----Removendo Paciente-----
Digite o CPF do Paciente que você deseja remover:
2
Paciente removido com sucesso!
APORTE ENTER PARA CONTINUAR
-----

```

Fonte: Autor (2023)

4.12. Caso 12 - Imprimir Consulta após Remoções

A opção número 7 imprime uma consulta específica ao ser indicado seu número específico (mostrado na funcionalidade do item 3.8):

Figura 17: Opção – Imprimir Consulta por Número após Remoções

```

7
-----Imprimindo Dados de Consulta-----
Digite o número da Consulta que você deseja ver as informações:
1
Consulta encontrada:
Número da Consulta: 1
Data: 20/09/2024
Hora: 15
CPF pessoas.Paciente: 2
CRM Médico: 123
Paciente de CPF 2:
(PACIENTE COM REGISTRO EXCLUÍDO)
Médico de CRM 123:
(MÉDICO COM REGISTRO EXCLUÍDO)
APORTE ENTER PARA CONTINUAR
-----

```

Fonte: Autor (2023)

Ao imprimir uma consulta após as remoções de seu médico e paciente, ela não é removida, porém indica que os registros foram apagados.

4.13. Caso 13 - Remover Consulta por Número

A opção número 9 solicita o número da consulta da qual o registro será apagado:

Figura 18: Opção – Remover Consulta

```
10
-----Removendo Consulta-----
Digite o número da Consulta que você deseja remover:
1
Consulta apagada com sucesso!
APERTE ENTER PARA CONTINUAR
-----
```

Fonte: Autor (2023)

4.14. Caso 14 - Imprimir Informações sem Cadastro: Médicos

Ao utilizar a opção número 4 novamente, será mostrada a seguinte mensagem:

Figura 19: Opção – Imprimir Médico

```
4
-----Imprimindo Dados de Médico-----
Não há médicos registrados!
APERTE ENTER PARA CONTINUAR
-----
```

Fonte: Autor (2023)

4.15. Caso 15 - Imprimir Informações sem Cadastro: Pacientes

Ao utilizar a opção número 6 novamente, será mostrada a seguinte mensagem:

Figura 20: Opção – Imprimir Paciente

```
6
-----Imprimindo Dados de Paciente-----
Não há pacientes registrados!
APERTE ENTER PARA CONTINUAR
-----
```

Fonte: Autor (2023)

4.16. Caso 16 - Imprimir Informações sem Cadastro: Consulta

Ao utilizar as opções de números 5 ou 7 novamente, serão mostradas as seguintes mensagens:

Figura 21: Opção – Imprimir Consultas

```
5
-----Imprimindo Lista de Consultas-----
Lista de Consultas vazia!
APERTE ENTER PARA CONTINUAR
-----
```

Fonte: Autor (2023)

Figura 22: Opção – Imprimir Consulta por Número

```

7
-----Imprimindo Dados de Consulta-----
Não há consultas registradas!
APERTE ENTER PARA CONTINUAR
-----

```

Fonte: Autor (2023)

4.17. Caso 17 - Cadastrar Consulta sem Médicos e/ou Pacientes Cadastrados

Ao utilizar a opção 3 para cadastrar uma consulta novamente, a seguinte mensagem será mostrada, caso não haja registro de médicos e/ou pacientes:

Figura 23: Opção – Cadastrar Consulta

```

3
-----Cadastrando Consulta-----
Não é possível criar uma consulta pois não há médicos ou pacientes cadastrados!
APERTE ENTER PARA CONTINUAR
-----

```

Fonte: Autor (2023)

4.18. Caso 18 - Sair do Programa

Ao utilizar a opção 11 o sistema é finalizado:

Figura 24: Opção – Sair do Programa

```

11
-----Saindo do Programa-----

Process finished with exit code 0

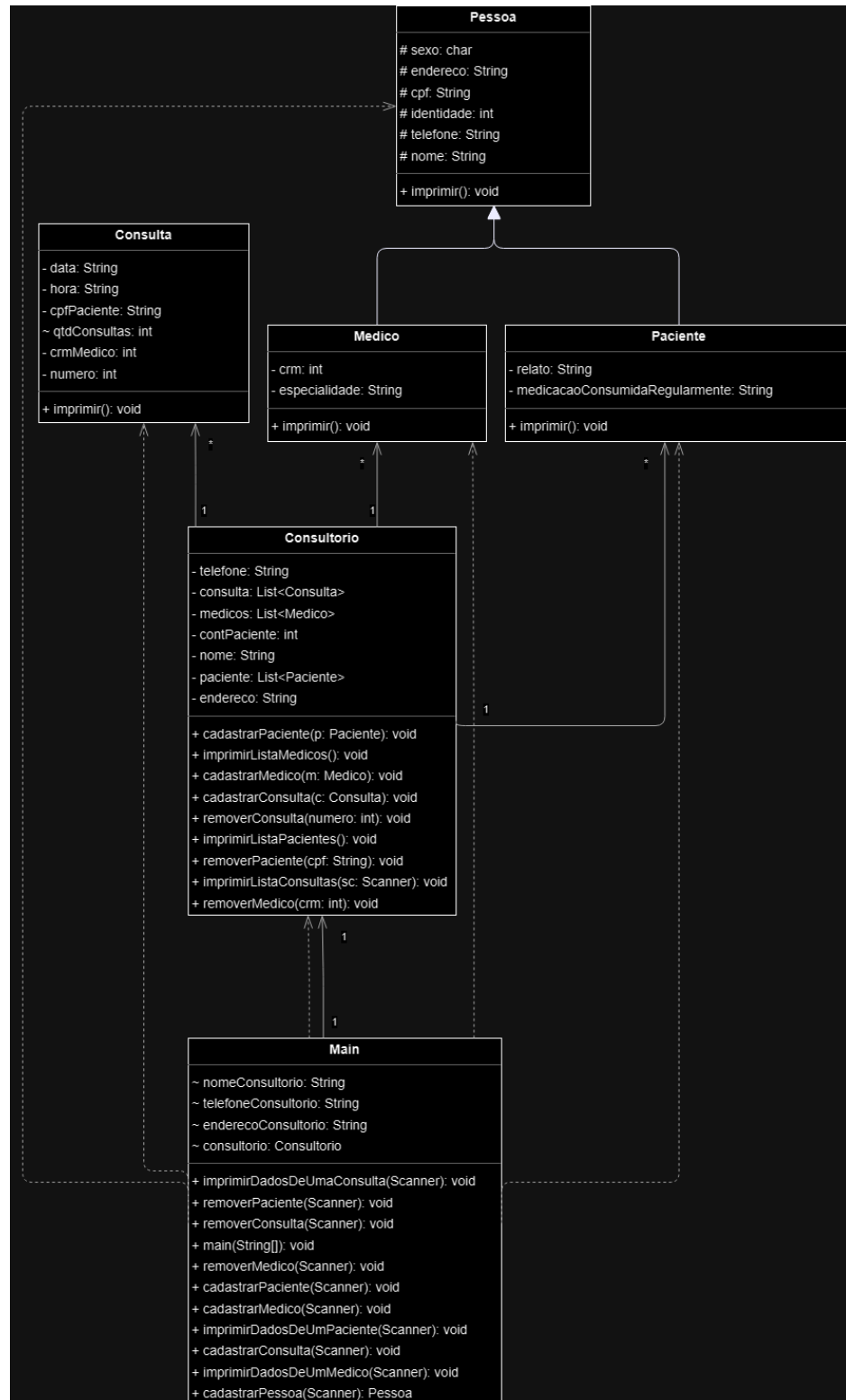
```

Fonte: Autor (2023)

5. ANÁLISE DOS RESULTADOS

Após a realização dos testes, foi possível perceber que as funcionalidades funcionam de acordo com o que foi inicialmente proposto, seguindo as especificações do diagrama de classes final:

Figura 25: Diagrama de Classes Final



Fonte: Draw.io – Autor (2023)